REMARKS

This Amendment is in response to the Office Action dated July 17, 2003. Claims 1-18 are pending. Claims 1-18 are rejected. Claims 1-2, 4, 8-10, 12, and 16-18 have been amended. Accordingly, claims 1-18 remain pending in the present application.

Claims 1-18 are rejected under 35 USC 102(b) as being anticipated by the article entitled, "An Advanced Commit Protocol for MLS Distributed Database Systems" by Indrajit et al. (hereinafter "Indrajit"). The examiner states:

> Referring to claim 1, Indrajit discloses a method for selectively releasing locks on data as claimed. See Figures 1-5 and the corresponding sections (1-5) of the article for the details of this disclosure. In particular, Indrajit teaches "a method for selectively releasing locks on data, comprising the steps of:
>
> (a) providing at least one savepoint [established by SaveWork() system call (See Figs. 1-5 & Section 4.1 in particular)] in a transaction [subtransaction or local transaction (See Fig. 2)], wherein at least one lock [low level lock9s) for op$_i$] is assigned to the at least one savepoint [See sections 4.1 – 4.2 (left column of pg. 123 in particular) & Algorithm 1 (in section 5)] and at least one lock [subtransaction locks other than low-read] is assigned to the transaction [See paragraph immediately preceding section 6];
>
> (b) rolling back [RollBack(sl) {See Fig. 2}] the transaction to the at least one savepoint [identified by handle sl]; and
>
> (c) releasing the at least one lock assigned to the at least one savepoint [See section 4.1 and the paragraph immediately preceding section 6], wherein the at least one lock assigned to the transaction is maintained [until commit (See the paragraph immediately preceding section 6)]" as claimed…
>
> Claim 8 is rejected on the same basis as claims 1-4 above. See the discussions regarding claims 1-4 for the details of this disclosure.
>
> Claims 9-15 are rejected on the same basis as claims 1-7 respectively. See the discussions regarding claims 1-7 above for the details of this disclosure. In particular, Indrajit's method described with regard to claims 1-7 above is implemented on a computer readable medium as claimed. See the Abstract, Introduction, and Section 2 for this disclosure.
>
> Claim 16 is rejected on the same basis as claim 8, in light of the basis for claims 9-15 above. See the discussions regarding claim 8 above for the details of this disclosure.
>
> Claim 17 is rejected on the same basis as claim 1. See the discussion regarding claim 1 above for the details of this disclosure. In particular, Indrajit's method described with regard to

claim 1 above is implemented on a system [distributed database system] as claimed.

Claim 18 is rejected on the same basis as claim 17. In particular, Indrajit teaches "a system, comprising:

> a data manager [transaction manager TM_s] for performing tasks [subtransactions] on data, wherein the tasks comprise:
>
> providing at least one savepoint in a transaction [claim 1, step (a)],
>
> assigning at least one lock to the at least one savepoint [claim 1, step (a)],
>
> assigning at least one lock to the transaction [claim 1, step (a)],
>
> responding to a recovery manager request for a rollback [claim 1, step (b)] to the at least one savepoint by undoing [See the 3rd paragraph of section 4.1] operations performed since the at least one savepoint according to a log [local transaction log (See section 4.1)], and
>
> requesting to a lock manager [See section 4.2] for release of the at least one lock assigned to the at least one savepoint [claim 1, step (c)];
>
> the lock manager [See section 4.2] for creating and maintaining the at least one lock assigned to the at least one savepoint and the at least one lock assigned to the transaction, and for releasing the at least one lock assigned to the at least one savepoint when the transaction is rolled back to the at least one savepoint [claim 1, step (c) and claim 4];
>
> a log manager [See section 4.1] for storing information on the tasks in the log [local transaction log], wherein the log comprises information concerning data and schema modifications, and the at least one savepoint [See Fig. 2]; and
>
> the recovery manager [Coordinator C], wherein the recovery manager processes the request for the rollback to the at least one savepoint by advising the data manager of the request [See Algorithm 1]" as claimed.

Applicant respectfully disagrees as to the claims as amended. The present invention, as recited in amended independent claims 1, 8, 9, and 16-18, provide a method, program instructions, and system for selectively releasing locks on data, comprising: (a) providing at least one savepoint in a transaction, wherein a first lock and a second lock are acquired after the at least one savepoint, wherein the first lock is assigned to the at least one savepoint and the second lock is assigned to the transaction; (b) rolling back the transaction to the at least one savepoint; and (c) releasing the first lock assigned to the at least one savepoint, wherein the second lock assigned to the transaction is maintained. By allowing.certain locks acquired after a savepoint to be assigned to the savepoint and other locks acquired after the savepoint to be assigned to the transaction, and releasing the locks assigned to the savepoint when a rollback to the savepoint occurs while the locks assigned to the transaction are maintained, these locks can be selectively released.

In contract, Indrajit discloses a rollback operation, which is an undo operation that releases any locks that op_i acquired on a data object, and remove the result of op_i from the system, as if opi was never executed. (See section 4.1) Indrajit further teaches that if a lower reading subtransaction decides to perform a roll back to a savepoint *s*, it release the locks acquired after the savepoint *s* and holds on to locks acquired before *s*. (See paragraph preceding section 6.) However, Indrajit does not teach assigning certain locks acquired after the savepoint to the savepoint and certain locks acquired after the savepoint to the transaction, such that the locks assigned to the savepoint are released in a rollback to the savepoint, while the locks assigned to the transaction are maintained. No such distinction in the locks acquired after the savepoint is made in Indrajit. Instead, in Indrajit, when a roll back to a savepoint occurs, all locks obtained after the savepoint are released.

Thus, Indrajit does not teach or suggest the combination of elements as recited in amended independent claims 1, 8, 9, and 16-18 of the present invention.

Therefore, for the above identified reasons, the present invention as recited in independent claims 1, 8, 9, and 16-18 is neither taught nor suggested by Indrajit. Applicant further submits that claims 2-7 and 10-15 are also allowable because they depend on the above allowable base claims.

In view of the foregoing, Applicant submits that claims 1-18 are patentable over the cited reference. Applicant, therefore, respectfully requests reconsideration and allowance of the claims as now presented.

The prior art made of record and not relied upon has been reviewed and does not appear to be any more relevant than the applied references.

Applicants' attorney believes this application in condition for allowance. Should any unresolved issues remain, Examiner is invited to call Applicants' attorney at the telephone number indicated below.

Respectfully submitted,
SAWYER LAW GROUP LLP

November 24, 2003
Date

Michele Liu
Attorney for Applicant(s)
Reg. No. 44,875
(650) 493-4540